Submitted by:
Bryan Rivas, Shahzeb Rizvi, and Yanwing Tsui

Advisor:
Dr. Zhigang Zhu

May 17th, 2021

Submitted in partial fulfillment of the requirements for senior design project

Computer Science Department
The City College of New York, New York, NY 10031

The City College of New York

Table of Contents

Introduction

Abstract

AlgoRythm is an education webapp which simplifies computer science concepts for high school students, middle School students and for individuals within the ASD spectrum specifically. Our aim is for our users to develop a computational way of thinking to make implementing algorithms in code a natural process. Our method is to teach algorithmic concepts at a high-level using animations and games, in essence this is the gamification of algorithms. When students play the games and watch the animations, they will be mastering algorithmic concepts subconsciously. This project was initiated to assist those under the ASD spectrum in learning computer science algorithms so that they may be able to find employment in software engineering related fields. As the project went along and after much research, our team concluded that this problem is much more uncharted than we once thought. There has not been enough research done in the field for us to provide even the least bit of an accurate estimation for the effectiveness of our approach. There was a slight pivot to start targeting students interested in the STEM field ages 12-16 since there has been extensive research on their learning habits and what are the best approaches to take. This market is much easier to penetrate helping AlgoRythm get proper footing. AlgoRythm intends to conduct research and continue to work with organizations that focus on individuals under the ASD spectrum so that we may in the near future develop our application to accommodate their needs. For We have developed a prototype with the input of individuals under the ASD spectrum, high school and middle school students, college students and even older individuals who would like to enter the tech field. This was our way of doing research and seeing if our method is sound. During each step of our development, we sent surveys and setup interviews with the users to make sure what we are developing makes sense. We were also concerned with seeing if our animations were actually able to teach students the concepts. To our surprise, they were, and performed even better than we initially thought. Students were able to learn concepts from the animations and apply them to the exercises we set

up with a little guidance. We will continue to improve our prototype to the point where we have developed a well-polished product which is ready to be deployed to schools throughout NYC.

Background

Autism Spectrum Disorder is a type of developmental diseases that can cause significant social, communication and behavioral challenges. According to the Centers for Disease Control (CDC) About 1 in 54 children has been identified with autism spectrum disorder (ASD) according to estimates from CDC's Autism and Developmental Disabilities Monitoring (ADDM) Network.[1] About 1 in 6 (17%) children aged 3–17 years were diagnosed with a developmental disability, as reported by parents, during a study period of 2009-2017.[1] These included autism, attention-deficit/hyperactivity disorder, blindness, and cerebral palsy, among others. The unemployment rate for people with ASD has been as high as 85%.[2] Out 18,000 people with ASD who used state funded vocational rehabilitation programs in 2014, only 60% left the program with a job.[3] However, 80% of the people that left with a job worked part time at a median weekly rate of $160.00.[3] Nearly half of the 25-year-olds with ASD have never held a paying job.[3] On the other hand the tech field is booming, especially now during this pandemic where most of our professional and even social interactions are all virtual. According to the U.S. Bureau of Labor Statistics, Employment in computer and information technology occupations is projected to grow 11 percent from 2019 to 2029, much faster than the average for all occupations. These occupations are projected to add about 531,200 new jobs.[4] The median annual wage for computer and information technology occupations was $91,250 in May 2020, which was higher than the median annual wage for all occupations of $41,950.[4]

Statement of the Problem

Individuals with ASD Vs Tech Field Employment

Due to the Covid-19 pandemic and now machines taking away even more jobs from workers, The U.S. lost over 60 million jobs—now robots, tech and artificial intelligence will take millions more.[5] This means that as a society we are moving towards being dependent on technology. This will cause many low skilled labor jobs to be automated leaving most engineering or tech related job open. This becomes an issue for individuals under the ASD spectrum who are already suffering from high rates of unemployment. If action is not taken it will be extremely difficult for those under the spectrum to find employment. The reason is that many under the spectrum are not given effective tools to learn technology. Many of them are very interested in the field however they are not given the chance.

Algorithms

Algorithms are a huge part of the tech field as it is used to optimize many solutions, which is why it is the best gateway to enter the field. Before we can truly understand the problem, we must know the difference between coding and algorithms. If an algorithm was a burger, then the ingredients of the burger would be coding. Just because one has learned coding does not mean they code a good algorithm, just like one who has never eaten nor seen a burger cannot put together a good burger.

Current Resources

The software engineering field is ever growing, however, to find employment in this field, one must be knowledgeable in Algorithms. Current resources on software engineering related fields are geared towards average intermediate to advanced college students. This means current books, videos, articles, tutorials, etc. are not beginner friendly and expect the viewer/reader to have knowledge of coding when coming in. Current resources are mostly focused on teaching coding first and then going into Algorithms. The problem with this approach is that mastering Algorithms requires a computational way of thinking. Most individuals are not naturally born with this and must be honed over the years, but as we get older it gets more and more difficult to change the way we think. This is where the issue comes in, by the time students are decent at coding, they are mindset is already too ripe, and it becomes difficult to learn algorithms. Therefore, if we start teaching algorithmic concepts to students from a younger age, by the time they start coding, algorithms will come naturally to them. This applies to both average students and students under the ASD spectrum.

Rationale of Solutions

Rationale

One of the most profound mysteries of Autism has been the remarkable ability of most Autistic people to excel at visual spatial skills while performing so poorly at verbal skills.[7] This form of intelligence, which Howard Gardner calls “spatial,” enables a person to base his or her method of learning on visual thinking instead of language-based thinking.[7] For visual thinkers, words are

like a second language. They can translate both spoken and written words into full-color movies, complete with sound which runs like a tape in their head.[7] Many parents of kids with Autism know very well that they became potty trained through picture related promptings. Visual learning is how they make sense of the world.[7]

In addition, computational thinking is a skill that is becoming a topic of increasing interest. The reason is due to the significant benefits associated with it in terms of problem solving.[6] Furthermore, a number of people in academia have come to realize the importance of bringing computational thinking to the core of many areas of study such as business and commerce, biology, and biomedical engineering.[6] There have even been suggestions brought up such as including computational thinking in preK-12 education.[8]

Lastly, many of the obstacles that we were facing decades ago when it came to bringing computational thinking to the classroom have disappeared. The access to computers and the skills to use a computer are mostly commonplace in today's world.[8]

Design and Development of Systems

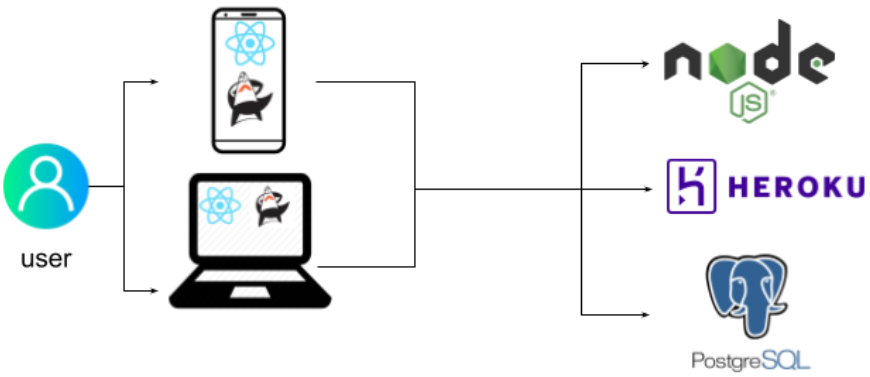

Figure 1. Technology stack used to implement AlgoRythm.

Technology Stack

We wanted to make an application that was accessible from any device with an internet browser, that is why we chose Ionic React as our front-end framework; the framework allows us to develop cross-platform applications with ease. Another advantage of Ionic-React is that its components are responsive out of the box, this allows us to focus on the design without worrying about how the app would look on devices of various screen sizes. Our animations are developed with a combination of CSS key-frames, React transition groups and Animatron, allowing us to create beautiful animations that are easily digestible for our users. For our Back-End framework we used Node JS which handles the data that our users might send. We implanted a RESTful API in Node.js with PostgreSQL as our database management system. We are hosting the Back end and database on Heroku as it is very easy to set up with no charge.

Learn About the Computer

Our app has a section that slowly introduces the user to what a computer is. The user can learn all the parts that a computer is composed of and how these parts interact with each other. This section was implemented for the very beginner that would like to know how the computer stores and displays information. In addition, by having access to this knowledge, the user can understand how data structures like arrays work at a more fundamental level, thus, creating less confusion on the later sections.

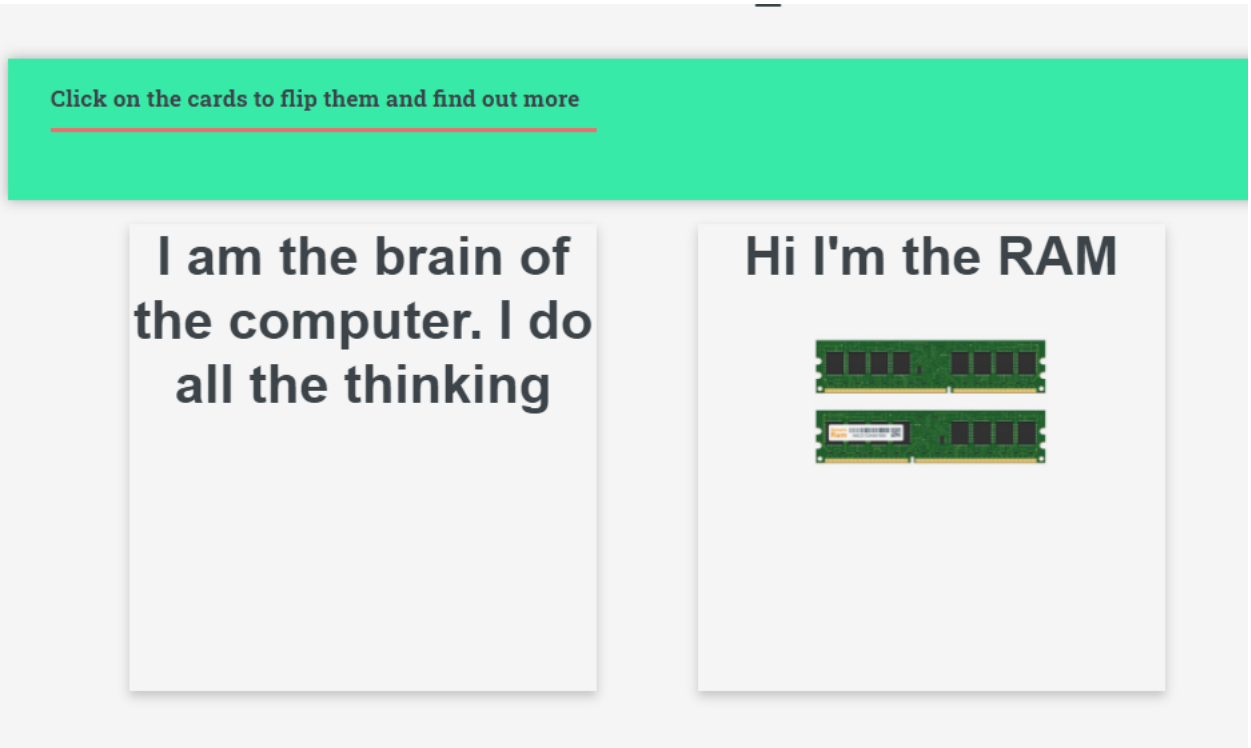

Figure 2. Showcases the learn about the computer section.

Learn About the Data Structures and Algorithms

This section consists of pages that contain the key aspects about each data structure or algorithm. Each page is written in a way that can be understood without prior computer science knowledge. We relate the applications of the concepts to real-world examples so that the user can better

understand its utility. In cases where the concept can be too ambiguous since the computer has limitations that a human does not, we make sure to explain the computer's limitations by comparing it to what a person would do and how said person would do it. This way it is clear for the user when something is done in a way that can be seen as inefficient for a "human". In addition, all the articles contain pictures, animations, or both. The written explanation introduces and explains the concepts in detail, while the picture/animation is displayed so that the users who are visual learners can take advantage of that.

Figure 3. Showcases the learn about the data structures and algorithm section.

Learn with Voiced Over Full Res Animations

Each of the data structures and algorithms that are covered by our application also includes a detailed but easily digestible voiced over animation that explains the ins and outs of the concept. We also made sure to include captions on each key moment for the hearing impaired. This is done for two reasons, number one, we wanted the users in the spectrum to be able to take advantage of their visual learning ability so that they can grasp as many details as they possibly can about the concept being explained. Including the details that are so subtle that only the visual learner might be keen enough to pick up. Number two, one of the aims of our applications is to simplify the learning process of these concepts as much as possible, some of the concepts can

become too confusing if we just rely on a written explanation so we went with the “show it” approach or as the saying goes a picture is worth a thousand words.

Figure 4.1 Full res animation for the queue data structure.

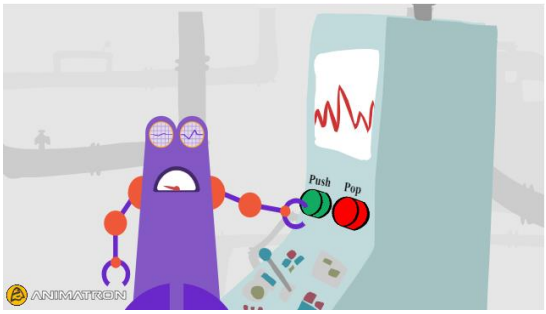

Figure 4.2 Full res animation for the stack data structure.

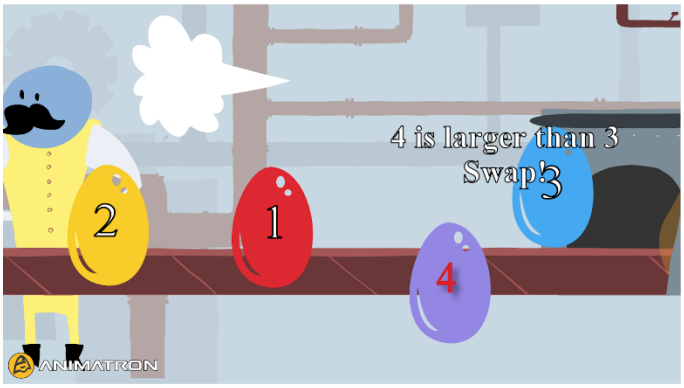

Figure 4.3 Full res animation for the bubble sort algorithm.

Practice Concepts with Interactive Examples

One of the principles that have been shown to increase learning and engagement is the tell, show, do and apply principle.[9] For this section in our application, we have created interactive

examples where our users can exercise the “do” principle so that they can solidify what they previously learned from “telling” and “showing.” These examples let our users interact with the data structures and practice the algorithms without having to write a single line of code. Some of the things our users can do are:

1. See how they can use the last-in first-out principle of a computer stack to reverse the order of things.
2. Practice algorithms like bubble sort by dragging and dropping the data that they are sorting in the same way that a computer would.
3. See how a queue is different from a stack because it follows the first-in first-out principle and see how it can be used to model things like a check-out line at the supermarket.

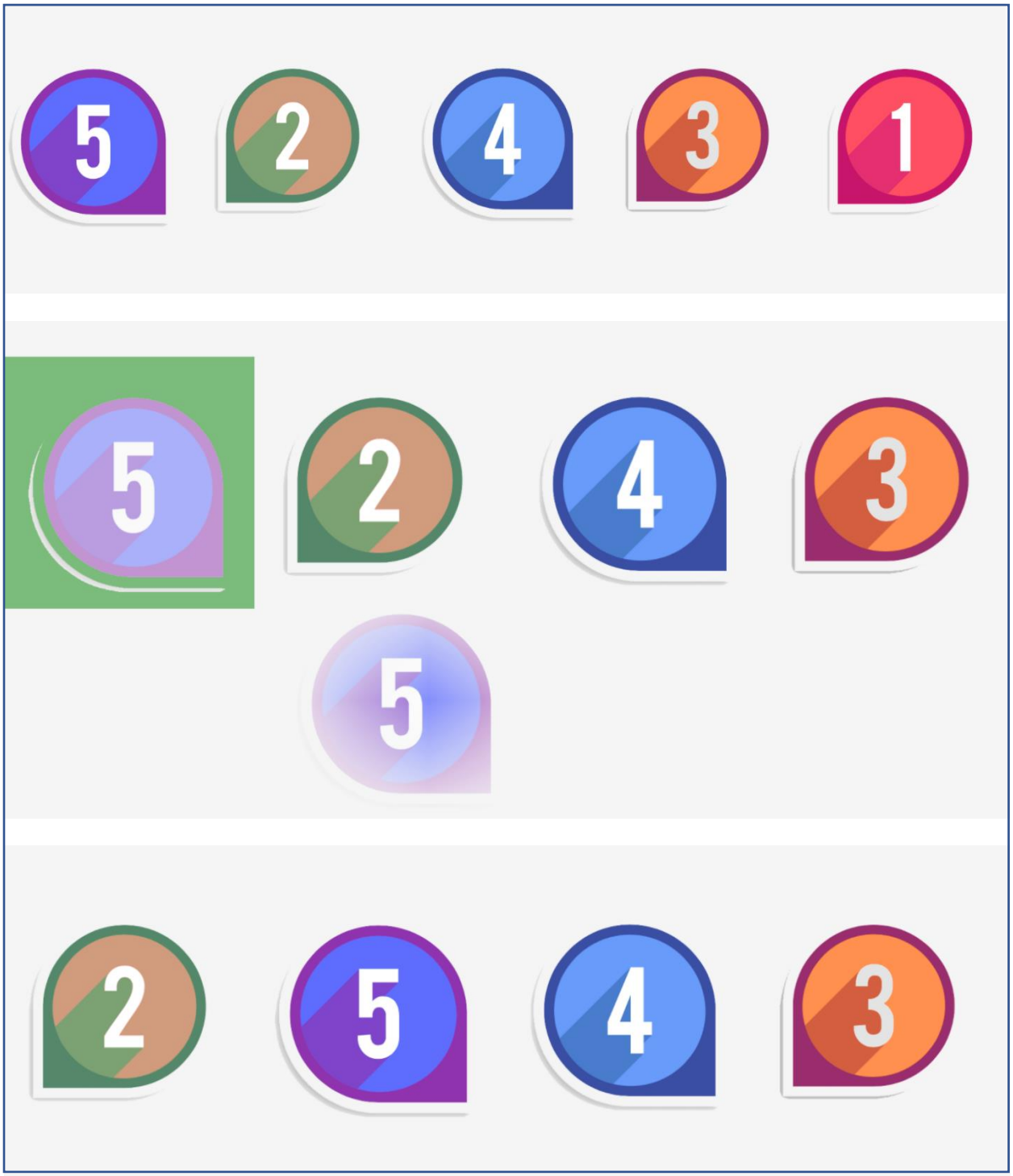

Figure 5. Bubble sort interactive example

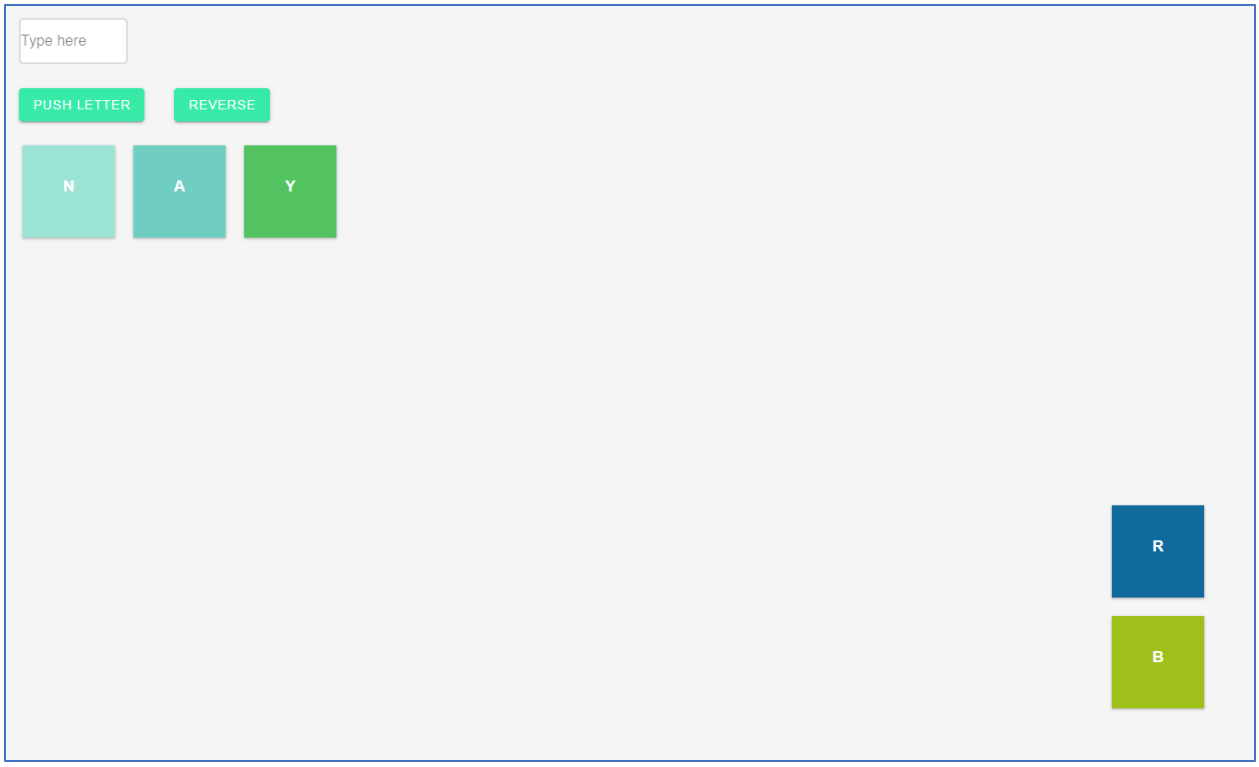

Figure 6. Stack interactive example (LIFO)

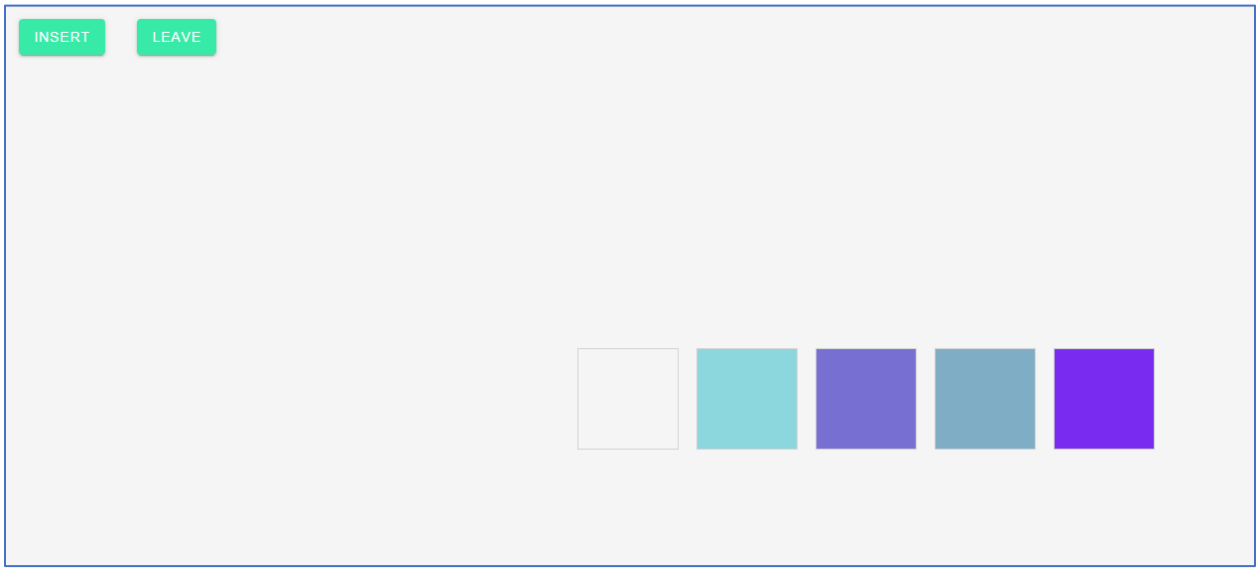

Figure 7.1 Queue interactive example. (FIFO)

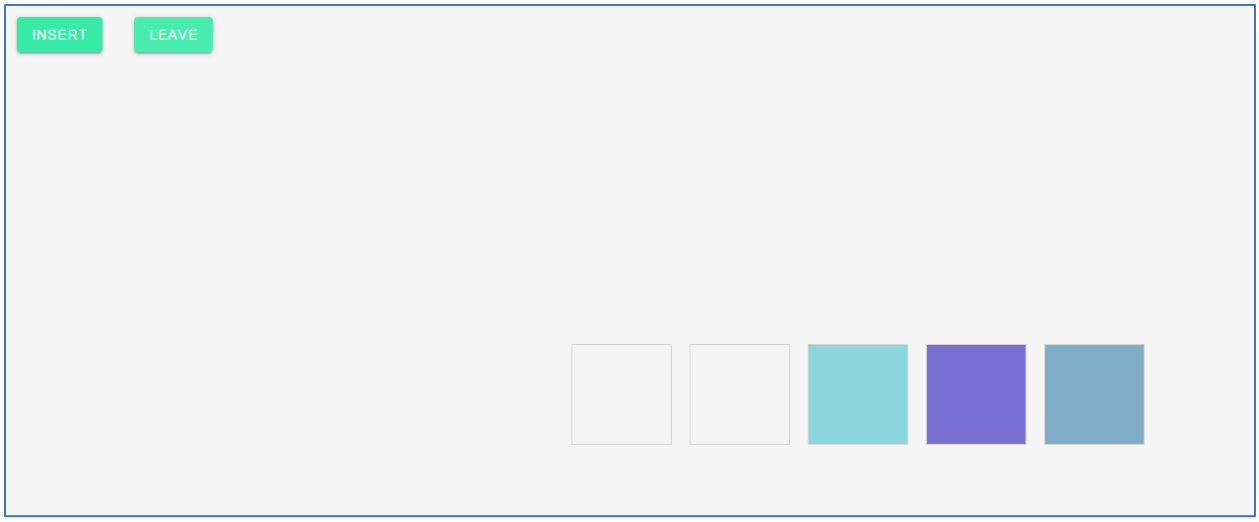

Figure 7.2 Queue Interactive example (Elements shifted to the right after the front element left the queue).

Challenge Yourself

Our application lets our user challenge themselves by practicing algorithms in a time attack fashion. The goal is for our users to beat their previous lowest time so that they can earn points that can then be exchanged for cool perks. In addition, they will also be placed on a leader board with other users so that they can have a sense of accomplishment and bragging rights. This was done to keep those with a competitive mentality motivated to learn and give those with a less competitive mindset a reason to try the challenges out.

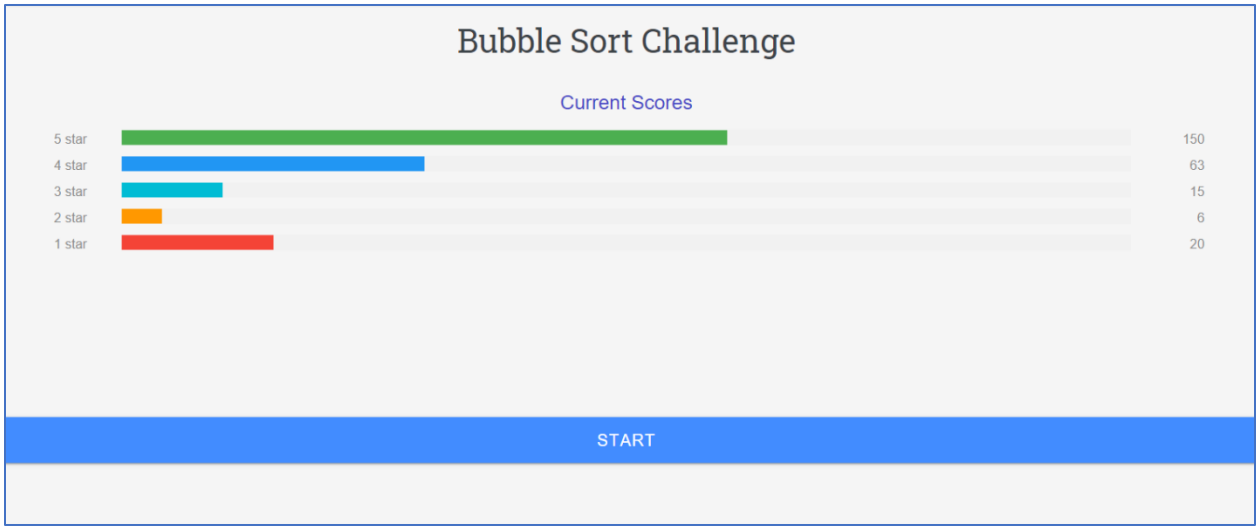

Figure 8.1 Challenge yourself screen

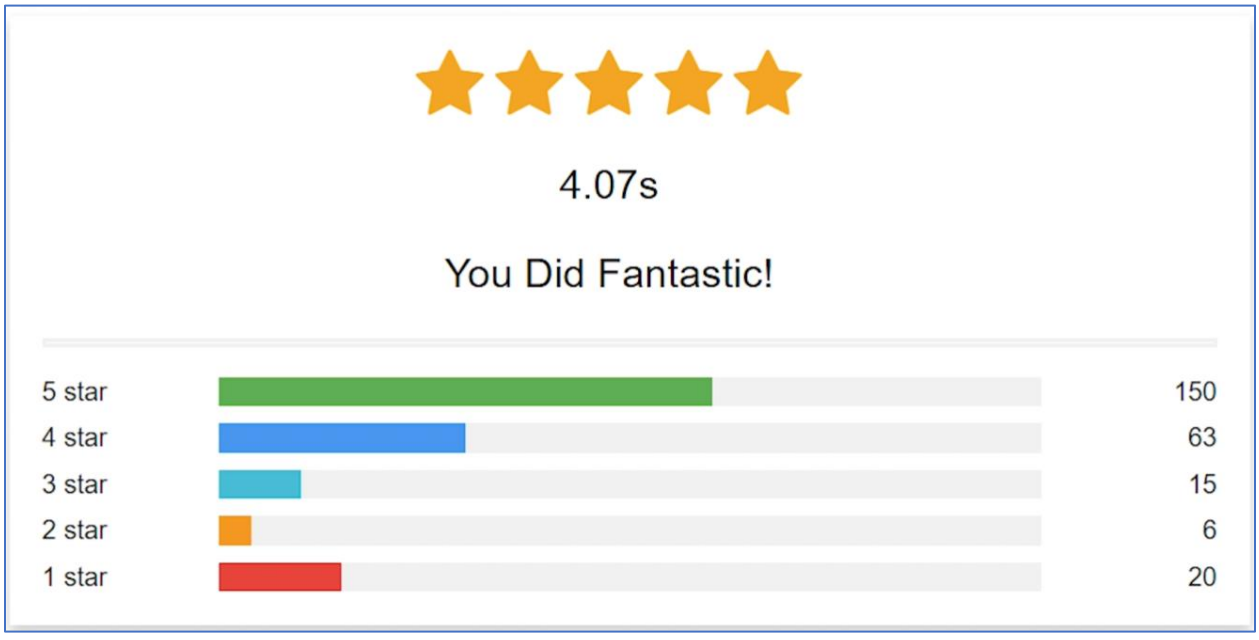

Figure 8.2. Completed challenge screen.

Notes

Reviewing lessons can be a tedious process, sometimes there is one key detail that is difficult to remember. For this reason, we implemented a notes section where our users can take notes of

whatever detail that they are having difficulty retaining. Users can quickly revisit those key details when they are trying to refresh their knowledge about a certain topic.

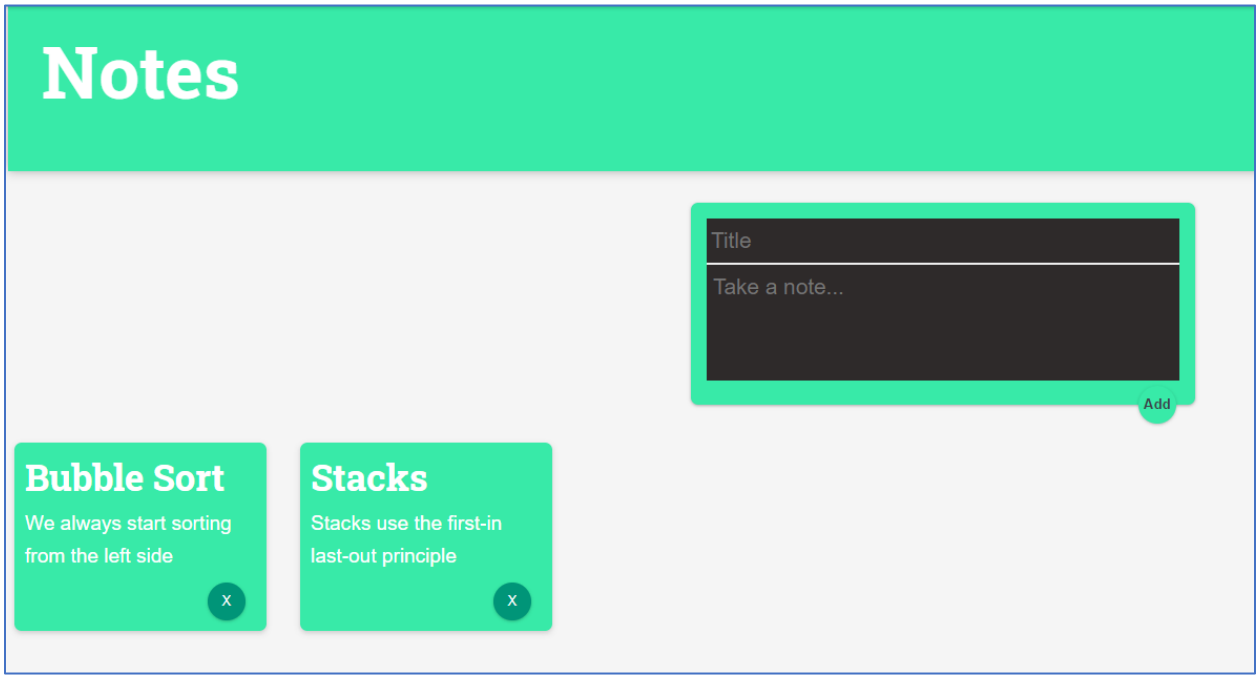

Figure 9. Showcases the notes section.

Evaluation with Users and Partners,

CCNY Computer Science Students

Last year, we started brainstorming for ideas to develop our app, the initial design was mainly inspired from the literature that we had read about the challenges that parents and educators face when teaching people with ASD. After we had our initial design implemented, we presented it to a set of judges and among those judges there were people that work directly with individuals under the ASD spectrum, and these judges provided invaluable feedback and guidance that helped evolve our idea.

Earlier this year, after developing the basic design of our app further with the feedback that we had received from the judges we decided to have CCNY students evaluate it for the design's feedback. As a result, we let over 20 CCNY students test the app and give us feedback on what they think should be

improved. One feedback we received was to add more color and styling to our app which students can find appealing. Therefore, we received guidance from a UX/UI design student from Brooklyn College that helped us redesign our menu to be better suited for our target audience. Another feedback we received was giving users the option to pick a theme. We thought this was a great idea since it will make our app more comfortable for different users. We implemented a theme switcher which switches between dark mode and light mode. The evaluation of our peers was critical in our initial design phase.

High School Students

Since our app is not only targeting neurodiverse students, but students in general, this year we interviewed more than 30 high school and middle school students while giving them a walkthrough of our app using the prototype that we had developed. The questions we asked covered everything from the design of the app to the specific concepts of algorithms and data structures which we are aiming to teach with our finished product. More than 85% of the students were able to grasp the concepts being taught. They were even able to understand the minor details that we thought we would have to explicitly convey for them to grasp. Even though we had a lot of success with our interviews, each student had good points in the minor feedback they gave us. We will do our best to address each of those concerns accordingly before we release our finished product.

Good Will

Professor Zhu introduced us to Tasha, who is a staff trainer from Goodwill, Goodwill is a non-profit organization that empowers people with disabilities through employment. We then touched base with her and scheduled a meeting back in April. The participants of the program were excited for the meeting however, due to COVID-19, we had to set it up virtually. The meeting consisted of adults who were under the ASD spectrum and with their coordinator. The meeting was amazing, all the participants were lively and were enthusiastic about our app. They even wanted to know when it was being deployed so that they may start using it. We received a lot of helpful feedbacks from them as well. One feedback that stuck out

in particular was that they wanted to know more about computers in general. We realized that in order for students to understand concepts like memory, they would need to start from the basics, such as the parts of a computer or how the system works. From that, we added an exercise which consists of flipping computer parts on a card to learn about them. We also learned from the interview that individuals under the spectrum often need to link concepts with reality. In other words, they want to be able to visualize the concepts in such a way where they can see it being applicable in their daily lives. We plan to make our games take place in settings that are familiar to students so that they can form those connections in their brains.

Discussion of Potential Markets and Future Work

Road Map

AlgoRythm is a three-phase project, where in each phase we expand and delve into a different and sometimes more complex market. We start with middle school and high school students in phase I, then in Phase II we expand to college students and finally in Phase III we expand to individuals with ASD. Let us go into each phase and the rationale behind them.

Phase I

Initially we are going to start with targeting average middle school and high school students. We realized there is a strong need for minorities to get in to computer science. Most often there is a late start for minorities in this field as they do not have the resources to get started with programming. By the time they reach college their counter parts are already getting on to more advanced topics and do better. The number of minority students taking computer science in High School is very low as well compared to other ethnicities. We want to tackle this challenge first. This is a very well-defined market with high

customer return rate as schools often buy licenses in bulk for their students. We were able to find a lot more statistics on STEM students. Teachers will also be able to use the app to track student's progress and give assignments based on what they see fit. Our aim is to have at least 50 schools that use our app in the next 3 years. If we take an average of 45 students that use our app per school and charge an average of $20 per student, we can make a profit of $240,000 which will break us even. During this time, we will be adding more and more concepts for students to learn.

Year	2022	2023	2024	2025	2026
Target schools	5	15	55	100	500
Quantity Sold	225	675	2475	4500	27000
Price per student /month	$10	$15	$20	$20	$20
Sales/ Revenue	$27,000	$121,500	$594,000	$1,080,000	$6,480,000
Net Revenue	-$181,344	-$47,714	$240,095	$509,306	$4,116,765

Phase II

From the profits we gain in Phase I we will be able to acquire a bigger team that will produce much more content. From here we can tackle much more advanced algorithmic concepts such as graphs and different tree algorithms. These concepts are well suited for college students since they are taking Algorithm and data structure classes. They will be able to learn these topics with a very simple approach. Also, we want to follow our clients in their entire algorithmic learning journey. In other words, we want to students to benefit from our app even after graduating from High School. This is where we implement the freemium service. The student can use the app and certain features for free. If the student used the application in High School, their progress would transfer over so that they continue where they left off. Instead of being

guided by their teacher, now in college they will be using the app independently. This will increase AlgoRythm's user base by almost double because in the USA alone, approximately 300,000 students are taking Computer Science classes, even if we reach 1% of that market, we will profit the same as if we had 55 more schools subscribing. From that increase in profit, we intend on hiring more staff and starting research on students with ASD in computer science. That staff will include animator who can produce inhouse animations for algorithmic concepts. Creating animations is not an easy task and is very time consuming, which is why we want to bring an expert in the field on board.

Phase III

Currently there is not enough research being done on individuals with ASD in computer science. This Is because there is a lack of focus from industry leaders on the subject. We want to be the pioneers who will include those under the spectrum and develop our app in such a way where they can benefit as well. This is our main goal for the app. We will be doing research and testing with ASD programs to help us make the perfect application. Already we are keeping many things in mind which include the font, colors and visuals we use. However, we are not able to back up any claims without statistics which can be acquired with experiments and testing. Accurate testing requires a good amount of financial backing and resources which is why it is important for AlgoRythm to start with Phase I and II. There are around 256,000 students with ASD in America's high schools [10]. There is a decent customer base as the rate of children with Autism is increasing every year.

Branding (the evolution of the names, logos and topics of the team project)

Name

The name “AlgoRythm” emphasizes one of the important programming concepts, algorithms. We took the first half of the word algorithm and combined it with a homophone of the word rhythm (*rythm*) to remind people that learning algorithms can be fun like playing an instrument. At first, you start by hitting random notes, but when you learn enough, you can find yourself playing wonderful songs while having lots of fun.

Logo

Our logo went through many iterations. Figures 10.1 and 10.2 are our initial designs, and figures 10.3 and 10.4 are our final designs. Figure 10.3 incorporates the main colors of our app. These colors were carefully chosen to represent trust and friendliness for newcomers, the cogwheel and wrench give a sense of technical refinement which is one of the core principals of our app. Figure 10.4 is a compact version of our logo in figure 10.3 that can be used in a splash screen or mobile app.

Figure 10.1 Design of our first logo

Figure 10.2 Second iteration of our first logo design

Figure 10.3 Design of our final logo

Figure 10.4 Compact version of our final logo

Typography

We also did quite a bit of research on typography, we found out sans-serif fonts like Montserrat and monospaced are some of the most well perceived fonts among dyslexic people.[10] Montserrat creates a majestic yet fun look which can grab user's attention during learning. Monospaced makes it easier to identify characters. Therefore, we will use these two typographies in our website.

MONTSERRAT

EXTRA BOLD

Monospaced

Main Page

#39eaa8 **#2e8bca** **#2962d9**

For the design of our main page, we used the colors #39eaa8, #2e8bca and #2962d9 because as we previously mentioned in the logo section, these colors represent trust and friendliness and we wanted to carry out that same feeling throughout the entirety of the app.

Figure 11. Showcases the main menu of the AlgoRythm app.

#720fcb **#220441**

We also implemented a dark theme for users that are sensitive to light or that simply do not like bright colors. The dark theme is made up of primarily two colors, #720fcb and #220441. Just as we did for our light theme, we chose colors that represent trust and friendliness, and also formality, since dark themes are often perceived as a more elegant look in other applications.

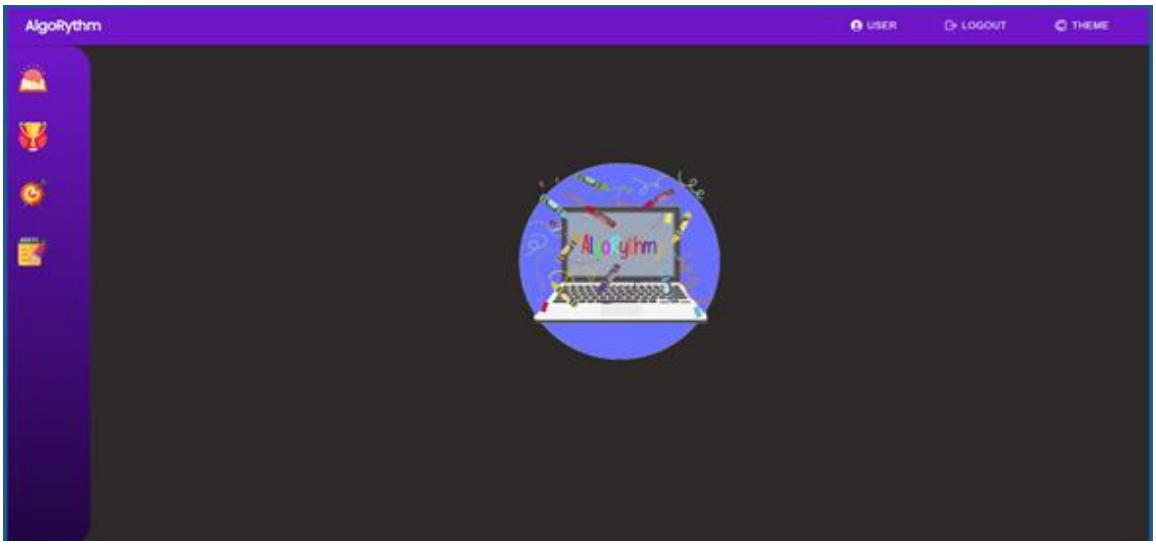

Figure 12. Showcases the dark theme of the AlgoRythm app.

References Cited

[1] *Data and Statistics on Autism Spectrum Disorder | CDC*. (2020, September 25). Centers for Disease Control and Prevention. https://www.cdc.gov/ncbddd/autism/data.html (accessed Nov. 2, 2020).

[2] Palumbo, J. (2021, April 27). *Why Autism Speaks Is Encouraging Companies To Hire Those On The Autistic Spectrum*. Forbes. https://www.forbes.com/sites/jenniferpalumbo/2021/04/27/why-autism-speaks-is-encouraging-companies-to-hire-those-on-the-autistic-spectrum/?sh=2460ba4852a2

[3] *"Autism and Unemployment."* linc-d.com. https://linc-d.com/about/the-statistics/#:%7E:text=Of%20the%20nearly%2018%2C000%20people,well%20below%20the%20poverty%20level (accessed Feb. 20, 2021)

[4] *"Computer and Information Technology Occupations*." U.S. Bureau of Labor Statistics. https://www.bls.gov/ooh/computer-and-information-technology/home.htm (accessed Oct. 10, 2020).

- [5] Kelly, J. (2020, October 27). *U.S. Lost Over 60 Million Jobs—Now Robots, Tech And Artificial Intelligence Will Take Millions More*. Forbes. https://www.forbes.com/sites/jackkelly/2020/10/27/us-lost-over-60-million-jobs-now-robots-tech-and-artificial-intelligence-will-take-millions-more/?sh=6a4ffed41a52

[6] Mohaghegh, Masha, and Michael McCauley. “Computational Thinking: The Skill Set of the 21st Century .” *International Journal of Computer Science and Information Technologies*, vol. 7, no. 3, 2016, pp. 1524–1530.

[7] Berube, C. T. (2007, May) from https://files.eric.ed.gov/fulltext/EJ967456.pdf. *TEACHING Exceptional Children Plus* [PFD]. Clair T. Berube.

[8] Thompson, A. D., Lindstrom, D. L., & Schmidt-Crawford, D. A. (2020). Computational Thinking: What Went Wrong? *Journal of Digital Learning in Teacher Education*, *36*(1), 4–5. https://doi.org/10.1080/21532974.2019.1696641

[9] Gardner, Joel. “Tell, Show, Do, Apply: The Anatomy Of Good Instruction.” *eLearning Industry*, eLearning Industry, 27 Mar. 2013, elearningindustry.com/tell-show-do-apply-the-anatomy-of-good-instruction.

[10] Rhodes, A. R. (2020, October 21). *What’s The Best Font For People With Dyslexia?* Extensis. https://www.extensis.com/blog/whats-the-best-font-for-people-with-dyslexia

Contributions

Logo design

- Initial Logo – Bryan Rivas
- Second Revision Logo – Bryan Rivas
- Third Revision Logo – Shahzeb Rizvi
- Mini Logo – Bryan Rivas

Customer Feedback

- Logo Feedback – Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Prototype Design Feedback - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Features Feedback - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Overall customer impressions of product - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Customer needs - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Setting up customer interviews - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui

Overall Idea

- Brainstorming - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Business Model - YanWing Tsui
- Scoping Competition - YanWing Tsui
- PowerPoint Design - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- User Stories - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Research - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui
- Developing Idea - Bryan Rivas, Shahzeb Rizvi, YanWing Tsui

Initial Prototype

- Setting GitHub – Bryan Rivas
- Setting up framework – Bryan Rivas, Shahzeb Rizvi
- Menu and header design – Shahzeb Rizvi
- Bubble Sort Animation – Shahzeb Rizvi
- Initial Bubble Sort game design – Bryan Rivas

Second Prototype

- User interface and menu redesign – Shahzeb Rizvi
- Bug fixes and page design – Bryan Rivas
- Menu tweaks – Bryan Rivas
- Refactored the source code – Bryan Rivas
- Implemented a note taking section – Bryan Rivas
- Implemented modal pop up for instructions – Bryan Rivas, Shahzeb Rizvi

Third Prototype

- Designed and implemented practice document pages – Bryan Rivas, Shahzeb Rizvi
- Digestible introduction to Stack data structure – Bryan Rivas
- Stack Practice game – Bryan Rivas

- Stack animation – Shahzeb Rizvi
- Stack animation voice over – Shahzeb Rizvi

Final Prototype

- Digestible introduction to Queue data structure – Bryan Rivas
- Queue Practice game – Bryan Rivas
- Queue animation – Shahzeb Rizvi
- Simple introduction to computer parts – Shahzeb Rizvi, Bryan Rivas
- User Profile Page - YanWing Tsui

Final Report

- Introduction – Shahzeb Rizvi
- Statement of Problem – Shahzeb Rizvi
- Rationale of Solutions – Bryan Rivas
- Design and Development of Systems – Bryan Rivas
- Evaluation with Users and Partners – YanWing Tsui
- Discussion of Potential Markets and Future Work – Shahzeb Rizvi
- Branding (the evolution of the names, logos, and topics of the team project) - YanWing Tsui
- References Cited – Bryan Rivas
- Acknowledgements – Bryan Rivas, Shahzeb Rizvi
- Revisions of report – Bryan Rivas, Shahzeb Rizvi

Acknowledgements

We would like to thank the support from the Zahn Center, Goodwill NYNJ, and The City College of New York. We would like to express our very great appreciation to Celina Cavalluzzi, Carina Aguilar, and Tasha Stanton from Goodwill NYNJ for user interviews and evaluation. We would also like to express great gratitude to our mentor Sam Scott who helped us refine our idea to the point at which it is today. Critiques and suggestions provided by the following judges are appreciated:

Anna Hutcheson (Zahn Innovation Center), Steven Monzon (Zahn Innovation Center), Katherine Olives (Zahn Innovation Center), Kesia Hudson (Zahn Innovation Center), and Hee Kim (Zahn Innovation Center).

We would also like to extend our thanks to Jose Mario for the mini logo design and to Shabih Rizvi for our menu design.

Last but not least we would like to give a special thanks to Professor Zhigang Zhu (The City College of New York) for giving us the motivation and opportunity to develop our idea. Without him this project would not be possible.